

Docket No. AUS920030468US1

**CLAIMS:**

What is claimed is:

1. A method of managing memory in a computing device, comprising:

receiving a notification of a runtime correctable error associated with a memory cell;

determining if the runtime correctable error has persisted for longer than one memory scrub cycle; and

requesting dynamic memory page deallocation for a page of memory associated with the memory cell with which the runtime correctable error is associated if the runtime correctable error has persisted for longer than one memory scrub cycle.

2. The method of claim 1, wherein determining if the runtime correctable error has persisted for longer than one memory scrub cycle includes:

comparing an address of the memory cell with addresses in a history correctable error table; and

determining that the runtime error has persisted for longer than one memory scrub cycle if the address of the memory cell is found in the history correctable error table.

3. The method of claim 2, wherein the history correctable error table is updated after each memory scrubbing operation cycle is completed.

Docket No. AUS920030468US1

4. The method of claim 2, further comprising:

determining if the address of the memory cell is present in a current CE table identifying memory cells that are have a correctable error identified in a current memory scrub cycle; and

incrementing a count for a matching entry in the current CE table if the address of the memory cell is present in the current CE table.

5. The method of claim 4, further comprising:

if the address of the memory cell is not present in the current CE table, determining if the current CE table has sufficient capacity for another entry;

adding an entry to the current CE table corresponding to the address of the memory cell, if the current CE table has sufficient capacity; and

setting a count associated with the entry to 1.

6. The method of claim 1, further comprising:

comparing a total number of correctable errors to a predetermined threshold; and

masking off further correctable error notifications until a current memory scrub cycle completes if the total number of correctable errors is equal to or greater than the predetermined threshold.

7. The method of claim 1, wherein receiving a notification of a runtime correctable error associated with a memory cell includes:

Docket No. AUS920030468US1

determining if there are any memory extents in the memory that have not been memory scrubbed;

if there are memory extents that have not been memory scrubbed, identifying a next memory extent that has not been memory scrubbed;

scrubbing the memory extent to identify any memory cell errors;

determining if an excessive number of memory cell errors occurred during the scrubbing of the memory extent;

determining if there are any redundant bit lines available for redundant bit line steering; and

sending the notification of the runtime correctable error if there are redundant bit lines available for redundant bit line steering.

8. A computer program product in a computer readable medium for managing memory in a computing device, comprising:

first instructions for receiving a notification of a runtime correctable error associated with a memory cell;

second instructions for determining if the runtime correctable error has persisted for longer than one memory scrub cycle; and

third instructions for requesting dynamic memory page deallocation for a page of memory associated with the memory cell with which the runtime correctable error is associated if the runtime correctable error has persisted for longer than one memory scrub cycle.

Docket No. AUS920030468US1

9. The computer program product of claim 8, wherein the second instructions for determining if the runtime correctable error has persisted for longer than one memory scrub cycle include:

instructions for comparing an address of the memory cell with addresses in a history correctable error table; and

instructions for determining that the runtime error has persisted for longer than one memory scrub cycle if the address of the memory cell is found in the history correctable error table.

10. The computer program product of claim 9, wherein the history correctable error table is updated after each memory scrubbing operation cycle is completed.

11. The computer program product of claim 9, further comprising:

instructions for determining if the address of the memory cell is present in a current CE table identifying memory cells that have a correctable error identified in a current memory scrub cycle; and

instructions for incrementing a count for a matching entry in the current CE table if the address of the memory cell is present in the current CE table.

12. The computer program product of claim 11, further comprising:

instructions for determining if the current CE table has sufficient capacity for another entry if the address

Docket No. AUS920030468US1

of the memory cell is not present in the current CE table;

instructions for adding an entry to the current CE table corresponding to the address of the memory cell, if the current CE table has sufficient capacity; and

instructions for setting a count associated with the entry to 1.

13. The computer program product of claim 8, further comprising:

fourth instructions for comparing a total number of correctable errors to a predetermined threshold; and

fifth instructions for masking off further correctable error notifications until a current memory scrub cycle completes if the total number of correctable errors is equal to or greater than the predetermined threshold.

14. The computer program product of claim 8, wherein the first instructions for receiving a notification of a runtime correctable error associated with a memory cell include:

instructions for determining if there are any memory extents in the memory that have not been memory scrubbed;

instructions for identifying a next memory extent that has not been memory scrubbed, if there are memory extents that have not been memory scrubbed;

instructions for scrubbing the memory extent to identify any memory cell errors;

Docket No. AUS920030468US1

instructions for determining if an excessive number of memory cell errors occurred during the scrubbing of the memory extent;

instructions for determining if there are any redundant bit lines available for redundant bit line steering; and

instructions for sending the notification of the runtime correctable error if there are redundant bit lines available for redundant bit line steering.

15. An apparatus for managing memory in a computing device, comprising:

means for receiving a notification of a runtime correctable error associated with a memory cell;

means for determining if the runtime correctable error has persisted for longer than one memory scrub cycle; and

means for requesting dynamic memory page deallocation for a page of memory associated with the memory cell with which the runtime correctable error is associated if the runtime correctable error has persisted for longer than one memory scrub cycle.

16. The apparatus of claim 15, wherein the means for determining if the runtime correctable error has persisted for longer than one memory scrub cycle includes:

means for comparing an address of the memory cell with addresses in a history correctable error table; and

Docket No. AUS920030468US1

means for determining that the runtime error has persisted for longer than one memory scrub cycle if the address of the memory cell is found in the history correctable error table.

17. The apparatus of claim 16, wherein the history correctable error table is updated after each memory scrubbing operation cycle is completed.

18. The apparatus of claim 9, further comprising:

means for determining if the address of the memory cell is present in a current CE table identifying memory cells that have a correctable error identified in a current memory scrub cycle; and

means for incrementing a count for a matching entry in the current CE table if the address of the memory cell is present in the current CE table.

19. The apparatus of claim 11, further comprising:

means for determining if the current CE table has sufficient capacity for another entry if the address of the memory cell is not present in the current CE table;

means for adding an entry to the current CE table corresponding to the address of the memory cell, if the current CE table has sufficient capacity; and

means for setting a count associated with the entry to 1.

Docket No. AUS920030468US1

20. The apparatus of claim 8, further comprising:

means for comparing a total number of correctable errors to a predetermined threshold; and

means for masking off further correctable error notifications until a current memory scrub cycle completes if the total number of correctable errors is equal to or greater than the predetermined threshold.

21. The apparatus of claim 8, wherein the means for receiving a notification of a runtime correctable error associated with a memory cell includes:

means for determining if there are any memory extents in the memory that have not been memory scrubbed;

means for identifying a next memory extent that has not been memory scrubbed, if there are memory extents that have not been memory scrubbed;

means for scrubbing the memory extent to identify any memory cell errors;

means for determining if an excessive number of memory cell errors occurred during the scrubbing of the memory extent;

means for determining if there are any redundant bit lines available for redundant bit line steering; and

means for sending the notification of the runtime correctable error if there are redundant bit lines available for redundant bit line steering.